DOCUMENT RESUME

ED 446 737 IR 020 366

AUTHOR Helic, Denis; Maurer, Hermann; Scherbakov, Nick

TITLE Authoring and Maintaining of Educational Applications on the Web.

PUB DATE 1999-00-00

NOTE 7p.; In: ED-MEDIA 99 World Conference on Educational Multimedia and Hypermedia & World Conference on Educational Telecommunications. Proceedings (11th, Seattle, Washington, June 19-24, 1999); see IR 020 357. For full text table of contents: http://www.aace.org/conf/edmedia/99/toca.pdf; individual papers: http://ww.aace.org/conf/edmedia/99/edm99add.pdf.

PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.

DESCRIPTORS *Authoring Aids (Programming); *Computer System Design; *Educational Technology; *Hypermedia; Models; *Multimedia Materials; *Programming; World Wide Web

ABSTRACT

This paper discusses problems concerning the design of educational applications in general, as well as specific problems connected with educational applications meant to be published on the Web. A new approach to the authoring and maintaining of Web-based educational applications is proposed which is based on the concept of hypermedia composites. Hypermedia composites can be seen as a higher level of hypermedia authoring; the hypermedia composite represents a collection of multimedia documents and/or other hypermedia composites. The Hypermedia Composite Data Model allows the possibility of defining many different "classes" of educational applications, to best match the requirements of a particular application. The architecture, components and functions of the Structure Editor, a system that implements the Hypermedia Composite Data Model, are described. Includes six figures. (AEF)

Reproductions supplied by EDRS are the best that can be made from the original document.

ED 446 737

Authoring and Maintaining of Educational Applications on The Web

Denis Helic
Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology Graz, Austria
e-mail: dhelic@iicm.edu

Hermann Maurer
Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology Graz, Austria
e-mail: hmaurer@iicm.edu

Nick Scherbakov
Institute for Information Processing and Computer Supported New Media (IICM), Graz University of Technology Graz, Austria
e-mail: nsherbak@iicm.edu

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY

G.H. Marks

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

Abstract: The presented paper clarifies the current situation at the field of the authoring and maintaining of educational applications on the Web. The paper lists well-known problems concerning the design of educational applications in general, as well as specific problems connected with educational applications meant to be published on the Web. Some comparison of widely used methods for authoring and maintaining of educational applications on the Web is, on the hand of method's advantages and disadvantages, presented as well. Thus, we propose a new approach to this specific problematic, that we believe, solves a number of problems related to the theme. This new approach to the authoring and maintaining of educational applications on the Web is based on the concept of hypermedia composites, so this concept we explain in details.

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

1. Introduction

An educational application can be seen as a special kind of a database system whose data contains some educational material. In modern educational applications the educational material is always presented using different types of media, such as text, graphics, animations, video, audio, and so on, i.e. an educational application is actually a kind of a multimedia database system. Further, an educational application has to insure a unique and non-sequential method of accessing information, and that leads us to the comprehension of the educational application as a special kind of a hypermedia system. The essential feature of a node-link model hypermedia system are documents nodes (HTML documents) and links. Documents can contain text, graphics, audio, video, animation, and images while links connect nodes related in a certain manner. It is the linking capability which allows the non-linear organisation of text. Currently, many organisations consider using hypermedia as an advanced educational media. Such educational hypermedia databases containing hundreds of documents are normally referred to as WWW educational applications.

The comprehension and general quality of an educational application in general, and a WWW educational application in particular, depends on the reader's ability to construct a coherent mental representation of the educational information. It is the author's responsibility to ensure the construction of the database as a coherent entity. The construction of a coherent hypermedia databases can be considered to be a design problem. There are no established guidelines for authoring such databases. Going beyond such well-known recommendations that a hypermedia database should consist of the following three components - the content part, the organisational part, and the presentation part.

Normally, authoring of a WWW educational application is carried out on a local authoring site, where a big number of HTML documents are created using such an easy to use WYSIWYG authoring environment as MS Front Page or Netscape Page Composer. After creating a sufficient number of documents they are interrelated by means of computer navigable links and the whole course is uploaded into a Web server where it becomes available for remote access.

While problem of authoring HTML documents got a scrutiny, and there even exists a number of solutions implemented as so-called HTML Editing Systems, the problem of navigating, or more precisely, of authoring of a convenient navigable structure helping users to construct a coherent mental representation of the educational information, does not attract much attention.

IR020366

BEST COPY AVAILABLE

Actually, when a particular author deals with creating hypermedia links (i.e. with imposing of a navigable structure on a top of a big number of HTML documents), this task is far from being a trivial one. First of all, educational applications are considered to be rather big ones. Further they are also heavily structured.
It is interesting to note that HTML authoring software generally do not use HTML tags as an authoring paradigm. Such authoring systems normally use a publishing logical model, where an author can place objects on a particular position, cut and paste fragments of arbitrary complexity, etc., At the same time, link editing still follows the most primitive node-link paradigm.

Obviously, decreasing of a interrelating complexity of a big number of HTML documents can be done via the usage of some powerful logical linking model (such as Hyperwave data model or HM-data model). This model should provide another logical view to hundreds of documents which should be interrelated, and thus make authoring considerable easier and even error-proven. But if we try to investigate all advantages and disadvantages when using some new logical data model in the authoring of educational applications, we see that such usage beside advantages, such as: considerably simpler authoring, support for referential integrity, to mention only the important ones has also a number of disadvantages. The disadvantages can be classified as follows: - authors need to learn a new data model, i.e. sometimes it needs a months to know all the facilities of a system supporting a new logical data model - the models support only primitive data structuring elements - the models do not reflect particular features of an application and still require tedious authoring.

Once, an educational application has been created, an author publishes it on the Web, i.e. the author uploads the educational application on a Web server. In the opposite to the authoring of WWW educational applications, which is done on a local site, the maintaining of WWW educational applications is mostly done on the server site, i.e. online. Of course, that in the case of some big changes that have to be performed on a WWW educational application an author can download the WWW educational application as a whole, modify it on a local site and then upload it once more, but that would rarely be the case, rather these changes are small and consist mostly of deletion or inserting of a small number of documents. But even in this simplest case the maintaining of WWW educational applications is connected with big problems. Let us here mention only few of them. If the node-link model has been used to prepare and publish a WWW educational application, i.e. the WWW educational application has been uploaded on a standard Web (HTTP) server the deletion of a document means also editing of all HTML documents pointing to the deleted one. In the case of the inserting of a HTML document we have the same problem. The usage of another, better structured, logical model can solve integrity and link consistency problems, but it has also earlier mentioned disadvantages.

Thus, we think that a new approach to the authoring and maintaining of educational applications on the Web should be introduced. We propose the concept of a hypermedia composites, that we believe solve a number of problems mentioned before.

Hypermedia composites should be seen as a higher level of hypermedia authoring. The hypermedia composite represents a collection of multimedia documents and/or other hypermedia composites. It has some internal navigational structure, that can be defined. The visualisation of a hypermedia composite can also be defined in desirable way. The hypermedia composite can be so organised that it full fills all needed aspects of a particular educational application. The data model which is the concept of hypermedia composites based upon could be classified as the semantic data model.

The semantic data models introduce purpose-oriented data structure types suitable for a particular application. These semantic data types with its navigational structure and its visualisation mechanism, as well with the data modification operations are defined using a well defined Data Definition Language (DDL) and Data Modification Language (DML) and are produced by the data administrator on the demand of an author. In this way an author can concentrate on the production of comprehend and convenient educational applications, so he can see the authoring process as in the first hand a design and not as a technical problem.

Each hypermedia composite can be mapped to a widely used hypermedia logical data model, in this way it could be uploaded on a Web server, and what is very important an inverse map can be performed, so the maintaining of a hypermedia composite, from an author point of view, does not differ from the authoring.

2. Hypermedia Composite

The hypermedia composite is a basic concept of what we called Hypermedia Composite Data Model. The Hypermedia Composite Data Model insures higher level of authoring and maintaining of educational applications on the Web. As object-oriented programming languages insure higher level of data abstraction then the procedural programming languages do, so the Hypermedia Composite Data Model gives us the possibility to define many different "classes" of educational applications, that will best match with the requirements of a particular application. Each "class" of educational applications has the predefined navigational structure and the visualisation paradigm. An author's task is to choose the best "class" of educational applications for his particular needs, then to construct a number of instances of this educational application class and fill it with HTML

documents. It is a task of a data administrator to produce "classes" of educational applications on specific demand of an author.

As we mentioned before the hypermedia composite is the main construct of the proposed model. Here we distinguish between two terms: the hypermedia composite unit (HC unit) and the hypermedia composite type (HC type). An HC unit represents a collection of HTML documents and/or other HC units, which are called members henceforth. Here we can draw a line of equivalency between an HC unit and an educational application, because an HC unit in the Hypermedia Composite Data Model is an educational application.
An HC unit can be treated in two ways. One way is the manipulation of the HC unit, i.e. an author can create a new HC unit, insert members into it, delete already existing members, save (publish onto a Web server) the HC unit, and so on.

The second way to treat an HC unit is to access it on the Web and to browse its content. This means that each HC unit, additionally, encapsulates a special navigational paradigm, i.e. computer-navigable links between members of the HC unit. As can be expected of a hypermedia system, whenever an user accesses such HC unit with an ordinary Web browser, it is visualised in a form of interrelated HTML pages. The concept can be explained with a simple example (Figure 1).

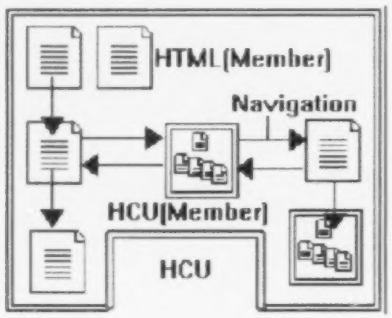

Figure 1: Hypermedia Composite Unit = Educational Application on the Web

Consider a hypermedia system that contains course modules. A set of nodes presenting information on a certain topic, can be joined together to form a conceptual group - an HC unit "Course with a given name". Thus the HC unit in question would contain HTML documents (members) - "title page", "abstract", "referential material", etc. Moreover, it might contain even other HC units presenting chapters units.

We can now generalise a number of HC units having the same navigational structure to a concept of the hypermedia composite type. A hypermedia composite type (HC type) is a meta definition of a specific linking structure which is automatically supported by all instances, i.e. HC units, of this type. It can be seen as an abstract data type and in the analogy to the programming languages as a class of objects. In this way we consider any HC unit as an instance of a particular HC type. Here we can say that an HC unit represents a class of similar educational applications, i.e. it is a template for the creation and manipulation of educational applications.

Thus we can say that a HC type automatically impose a particular navigable structure on the top of collection of existing HTML pages r other HC units defined as members of a HC unit. For example we can define an HC type "Course" (Figure 2).

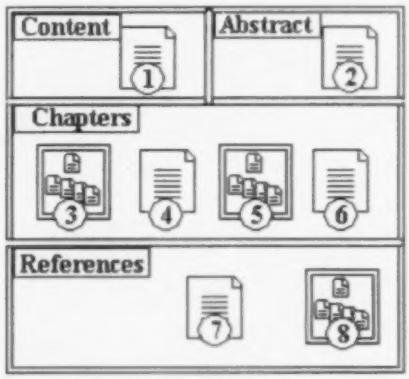

Figure 2: Hypermedia Composite Type = Class of Educational Applications

Any instance of this type is an HC unit consisting of HTML documents (or other HC units) labelled as: "Content", "Abstract", "Chapters" or "References". The term "labelled" deserves an additional discussion since it is very important concept of HC types.

In fact, members of an HC unit play essentially different roles when such unit is accessed or browsed by users. In our particular case, users might expect that: - the document "Content" is shown and provide references to all "Chapters" whenever they access the course; - any "Chapter" is provided with references to the "next" and "prior" "Chapters"; - "Chapters" are automatically provided with a number of "References" for further diting, etc.

These "labels" are used by the mapping mechanism in order to get the proper visualisation of an HC unit when it is accessed or browsed, For example if an HC unit is to be mapped onto the Hyperwave logical data model, most probably "labels" will be mapped into the attributes of a Hyperwave object, so they can be later interpreted in the proper way. More on this topic in the following chapters let us now go back to the concept of HC types.

Practically speaking, we can perceive an HC type as a special template consisting of a number of cells. Each cell represent a member (a set of members) having identical properties. Similarly an HC unit might be seen as an HC type template filled with existing HTML documents and/or other existing HC units.

Thus, from an author point of view, there is a number of predefined templates (HC types) where the author can simply insert existing pages or other HC units to define sophisticated navigable structure. Of course if an author has in the mind a special linking structure that has not been defined yet, he can ask a WBT administrator to create an HC type, i.e. an HC type template that comprises wished linking structure.

In the following chapter we present an implementation of the proposed model which is a running project on the Institute for Information Processing and Computer Supported New Media, called the Structure Editor.

3. Structure Editor Architecture and Components

The Structure Editor is a system implementing proposed Hypermedia Composite Data Model. The system is used for the authoring and maintaining educational applications on the Web. It treats educational applications as HC units which can be created, manipulated, uploaded onto a Web server, maintained on the server and so on. It also provides a data administrator with the tools for the definition of HC types, i.e. templates for the creation of different educational applications. Let us now look closely on the architecture of Structure Editor and its components.

The Structure Editor consists of the following functional components (see Figure 3):

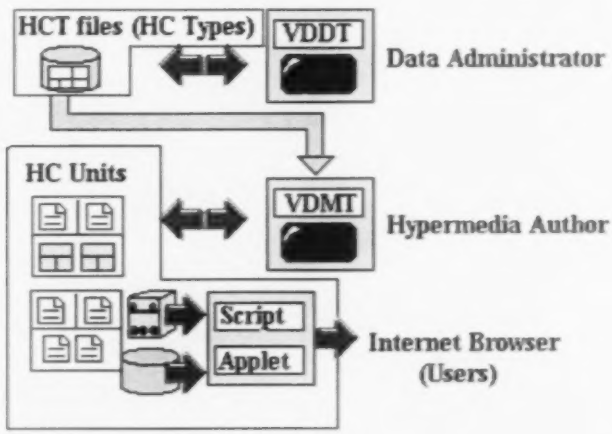

Figure 3: Structure Editor Components

- Visual Data Definition Tool (VDDT) provides a convenient way for defining HC types, i.e. templates for different educational applications. The definition of HC types means the definition of topology of templates, functionality and properties of individual cells, a specific linking structure which is inherited by all instances of this type, i.e. by all educational applications produced with this template.
- Visual Data Manipulation Tool (VDMT) provides a convenient way for automatic generation of a navigational structure of an educational application by means of inserting/removing elements into/from HC type template, saving such units on a Web browser (it could be a standard Web server or a Hyperwave server, i.e. the mapping mechanism will upload an educational application in the proper way) or on a local drive and further editing, maintaining of the existing HC units, i.e. educational applications.

- Server Site Script/Remote Applet are special programs which visualise a particular educational application as a collection of interrelated HTML documents.

The Visual Data Definition Tool is a stand-alone Java application working with a special HCT file containing the definition of a HC type. It can create new and edit existing HCT file residing on a local drive.

The Visual Data Manipulation Tool is a stand-alone Java application working with an existing HCT file, HTML pages and existing HC units, i.e. educational applications. It can create new and edit existing educational applications residing on a Web server or on a local drive. The main way of authoring and maintaining such educational applications is dragging and dropping existing HTML documents, MM elements and/or HC units onto a selected template. HTML documents can be created with an arbitrary other editing system and must be in a valid HTML format. Additionally, the structure editor provides a possibility to generate valid HTML documents on the fly using so-called Macros defined as a part of the HC type definition.

The VDMT allows to combine pages and existing educational applications into a new educational application by means of the following operations:

- drag and drop objects into a template cell
- delete an object from a template cell
- change a relative position within a template cell
- set up particular member attributes (see Figure 4)

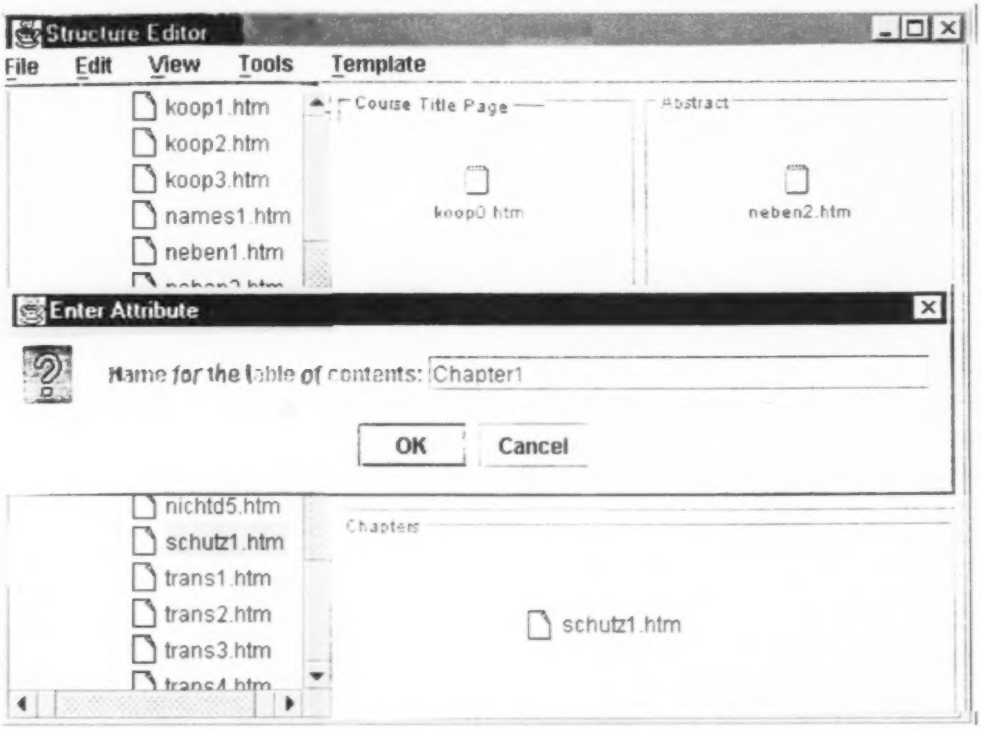

Figure 4: Working with the VDMT

A newly created educational application can be stored, i.e. mapped on a Hyperwave server, a standard HTTP Web server or on a local drive. Storing an educational application to a particular hypermedia system means providing it with some additional attributes specific to the chosen system. For example if an educational application is stored on a Hyperwave server, it is stored as a Hyperwave collection having a number of additional attributes which are automatically assigned by the VDMT. Members of the HC unit are defined as members of the corresponding Hyperwave collection and also automatically provided with specific set of attributes.

An educational application can be also stored on a local drive or on an ordinary Web server as a directory in a file system containing a number of additional attribute files which are automatically generated by the VDMT. Members of the HC unit are also put into the corresponding folder along with automatically generated attribute files and a special navigational applets interpreting the attribute files.

As stated earlier user's Web browsers (like Netscape or Microsoft InternetExplorer) do not access directly an educational application. Instead, in order to obtain data, the browser communicates with a special Sever Site Script (if the educational application resides on a Hyperwave server) or with a special Java applet (if the educational application resides on a standard HTTP Web server or on a local drive).

BEST COPY AVAILABLE

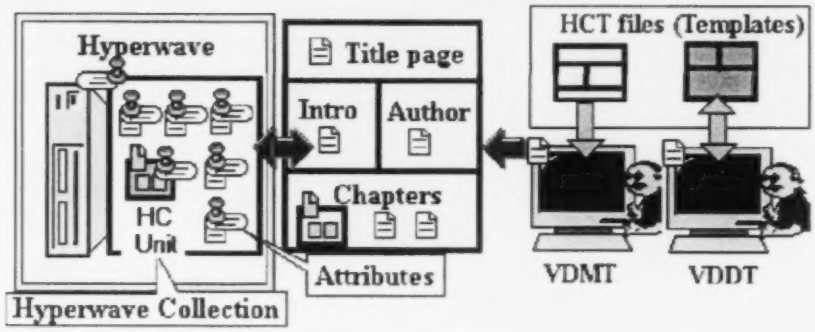

Figure 5: Saving an Educational Application on a Hyperwave Server

In other words, whenever an user accesses such educational application with an ordinary Web browser a special software component is run to visualise the unit in the form of interrelated HTML pages. The HCT file containing a description of the generic link structure and attributes attached to the educational application's members are essentially used to control such visualisation (see Figure 6).

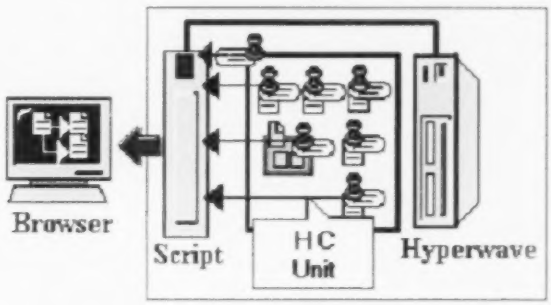

Figure 6: Accessing an Educational Application

4. Conclusion

Here, we would like to conclude that the proposed combination of Hypermedia Composite data model with Hyperwave data model in order to decrease effort for authoring and maintaining educational applications for the Web has much bigger possibilities then the described use. It can be very useful for different kinds of hypermedia systems. The power of this concept comes from the highly structured hypermedia composites that can be created with Hypermedia Composite data model. Of course that Hypermedia Composite data model has to be mapped onto a logical data model to show all its possibilities so we believe that its use should not only depend on Hyperwave data model, but rather it should try to use other powerful data models, such as XML. XML technology allows the author to define the constructs, i.e. the structure of hypermedia database, as well as the navigational and visualisation paradigm. Hypermedia Composite data model on the other hand allows the author to automatically and rapidly create predefined structures.

5. References

1. Hermann Maurer (1996), *Hyperwave – The Next Generation Web Solution*, Addison-Wesley

2. Denis Helic, Seid Maglajlic, Nick Sherbakov (1999), Educational Materials on The Web: Data Modelling Approach, *MIPRO 99*

3. D. Freismuth, K. Schmaranz, B. Zwantschko (1998), *Telematic* Platform for Patient Oriented Services, *JUCS, vol. 11, 1998*

4. Microsoft Online XML Workshop (1998), http://www.microsoft.com/xml, *Microsoft Corp. Homepage*